Model-based Inference of Haplotype Block Variation

Gideon Greenspan
Computer Science Department
Technion
Technion City
Haifa 32000
Israel
gdg@cs.technion.ac.il

Dan Geiger
Computer Science Department
Technion
Technion City
Haifa 32000
Israel
dang@cs.technion.ac.il

ABSTRACT

The uneven recombination structure of human DNA has been highlighted by several recent studies. Knowledge of the haplotype blocks generated by this phenomenon can be applied to dramatically increase the statistical power of genetic mapping. Several criteria have already been proposed for identifying these blocks, all of which require haplotypes as input. We propose a comprehensive statistical model of haplotype block variation and show how the parameters of this model can be learned from haplotypes and/or unphased genotype data. Using real-world SNP data, we demonstrate that our approach can be used to resolve genotypes into their constituent haplotypes with greater accuracy than previously known methods.

Categories and Subject Descriptors

J.3 [**Computer Applications**]: Life and Medical Sciences—*biology and genetics*

General Terms

Algorithms, Performance, Experimentation

Keywords

haplotype, haplotype block, recombination hotspot, haplotype resolution, Bayesian Network, linkage disequilibrium mapping

1. INTRODUCTION

Several recent studies suggest that the relationship between the physical distance separating loci on human chromosomes and the probability of their division during recombination is far from smooth [1, 2, 3, 4, 5]. Specifically, there are indications that in certain chromosomal regions, recombination only takes place at narrow hotspots, which separate between stretches of DNA which are almost never themselves divided during meiosis. The variants of these stretches, called *haplotype blocks*, constitute the true co-segregating alleles.

The visibility of haplotype blocks is enhanced by the low level of variation present within each, due to bottleneck effects and genetic drift. Bottlenecks occur when a local population is descended from a small group of individuals, for example due to migration or strong selection, resulting in a sharp reduction in genetic variation. Genetic drift refers to the gradual decrease in variation due to repeated random sampling of the alleles in a population from those in the previous generation. Since genetic drift is strongest when a population is small, the early generations following a bottleneck event will undergo the greatest reduction in diversity, leaving behind a small number of ancestral haplotypes upon which the future population is built.

The identification of haplotype blocks improves the effectiveness of the linkage disequilibrium (LD) approach to genetic mapping. The LD method is based on the assumption that the genetic variants underlying a disease are the product of mutations which took place in only a few founding individuals. Any marker allele possessed by one of these founders which is located in the same haplotype block as the disease allele will be passed together with the other alleles in that block to future generations. Therefore, the presence of the disease in affected individuals will be correlated with that marker allele, allowing the gene affecting the disease to be mapped. Knowing the haplotype block structure of a chromosomal region allows tests to be performed on multiple adjacent markers belonging to each block, dramatically increasing the chance of detecting associations.

Several tests have recently been proposed for detecting haplotype blocks in DNA. Daly *et al.* [3] identify stretches which have significantly less heterogeneity than would be expected considering the frequencies of the constituent SNPs. Patil *et al.* [4] and Zhang *et al.* [6] examine the ratio between the number of SNPs in a region and the size of the smallest subset of these which is sufficient to uniquely identify all of its haplotypes. Gabriel *et al.* [5] look for areas within which the allelic correlation between most pairs of SNPs is high. All of these criteria are local in that an assessment of each putative block is based only on the haplotype distribution observed within.

A potential obstacle for both haplotype block identification and LD mapping in general is the cost involved in separately identifying the complete haplotypes on each of a subject's two chromosomes. In the absence of additional

RECOMB'03, April 10–13, 2003, Berlin, Germany.

information from relatives, a standard genotyping process will yield an unordered pair of alleles for each locus, with no information on which alleles are co-located on the same chromosome. Molecular laboratory techniques to identify chromosomal haplotypes have been developed [7, 8, 9, 10] but their cost remains prohibitive in many cases.

A series of observed marker pairs containing s heterozygous sites can be separated into constituent haplotypes in 2^{s-1} different ways. This degeneracy leads to the haplotype resolution problem, which seeks to infer the pairs of haplotypes from which a set of observed genotypes are constituted. An early approach to haplotype resolution was Clark's parsimony-based algorithm [11], later improved by Gusfield [12]. A likelihood-based EM algorithm [13, 14, 15] gives far superior results but is infeasible for large experiments, since for genotypes with s heterozygous loci its complexity is $O(2^s)$. Recently, Stephens *et al.* [16] and Niu *et al.* [17] have proposed new MCMC-based methods which are computationally feasible and give good results. None of these methods for haplotype resolution consider the implications of the block-like structure of DNA.

We have developed an integrated approach to both haplotype block identification and haplotype resolution, suitable for high-density SNP data. It is based on a statistical model which takes account of recombination hotspots, bottlenecks, genetic drift and mutations. We show how the parameters of this model can be recovered from observed haplotype or genotype data and demonstrate the effectiveness of our technique by applying it to the haplotype resolution problem. For high-density regions of chromosome 21, our site pairwise error rates are between 3 and 200 times lower than those achieved by previously published methods.

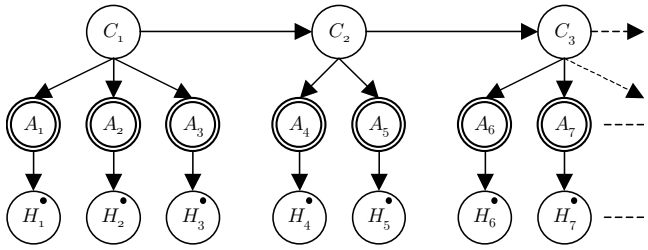
The rest of this paper is organized as follows. Section 2 describes our statistical model and its parameters. Section 3 explains the criterion used to assess how well a particular model fits some observations. Section 4 outlines the algorithm we use to search for a model which optimizes this criterion. Section 5 explains how a specific model can be applied to perform haplotype resolution. Section 6 compares the results of applying our approach in this way against existing methods for haplotype resolution. Finally, section 7 describes some future directions we aim to pursue.

2. STATISTICAL MODEL

Our model for the distribution of haplotypes descended from a bottleneck event can be represented as a *Bayesian Network*. A Bayesian Network is a directed acyclic graph, where each vertex $v = 1 \ldots n$ corresponds to a discrete variable X_v and each directed edge represents conditional dependencies between these variables [18, 19]. The distribution for each variable X_v is conditional upon the variables in Pa_v, which is defined as the set of vertices from which there are edges leading to v in the graph. The joint probability of a full assignment $x_1, \ldots, x_n$ to variables $X_1, \ldots, X_n$ is the product of these conditional probabilities. In other words, $Pr(X_1 = x_1, \ldots, X_n = x_n) = \prod_v Pr(X_v = x_v | Pa_v = pa_v)$, where pa_v is the joint assignment $\{x_i | X_i \in Pa_v\}$ to the variables in Pa_v. From here on, we will use the notation $Pr(y|z)$ as an abbreviated form of $Pr(Y = y|Z = z)$ for any sets of variables Y and Z. For example, the joint probability could be rewritten as $Pr(x_1, \ldots, x_n) = \prod_v Pr(x_v|pa_v)$.

An important query is to compute the probability of a partial assignment x_s to variables $X_s \subseteq \{X_1, \ldots, X_n\}$. This is

Figure 1: Bayesian Network for haplotype data



defined as the sum of $Pr(x_1, \ldots, x_n)$ over all full assignments $x_1, \ldots, x_n$ which are compatible with x_s, so that $Pr(x_s) = \sum_{x_1} \cdots \sum_{x_n} Pr(x_1, \ldots, x_n | x_s)$. The independence assumptions embedded in the Bayesian Network allow such computations to be performed efficiently, for example by bucket variable elimination, a technique applied extensively in our work [20]. Also, suitable parameters for the conditional distributions in a Bayesian Network can be learned from observed data sets by the Expectation Maximization (EM) algorithm, which we use at many stages during our search for a model to fit observations [21].

An example of our model is shown by the Bayesian Network in Figure 1. It consists of a random variable C_k for each block $k = 1 \ldots b$ and two random variables A_j and H_j for each SNP $j = 1 \ldots l$. Variable C_k takes values $1 \ldots q_k$, where q_k specifies the number of different haplotypes for block k which emerged from the bottleneck event, hereafter referred to as the *ancestors* for block k. Both A_j and H_j take values from the set B of SNP alleles, where $B = \{A, C, G, T, -\}$ contains the four nucleic acids and a deletion. A partition by recombination hotspots of the SNPs into blocks is defined by the groups of variables A_j pointed to by each C_k in the Bayesian Network. For example, the model in Figure 1 places hotspots between adjacent SNP pairs 3–4 and 5–6.

An assignment of values to the variables in the Bayesian Network reflects the history of a single observed haplotype. The value of each variable C_k is the index of the ancestor for block k from which the observed haplotype is descended. The sequence of that ancestor is specified by the values of $A_{s_k} \ldots A_{e_k}$, where A_{s_k} and A_{e_k} are the first and last variables descended from C_k respectively. The observed haplotype is specified by the values of variables $H_1 \ldots H_l$. Clearly, $H_j = A_j$ unless a mutation has taken place at site j since the bottleneck event.

The topology of the Bayesian Network defines the joint distribution $Pr(c_1, \ldots, c_b, a_1, \ldots, a_l, h_1, \ldots h_l)$ over all variables as:

$$Pr(c_1) \prod_{k=2}^{b} Pr(c_k|c_{k-1}) \prod_{k=1}^{b} \prod_{j=s_k}^{e_k} Pr(a_j|c_k) Pr(h_j|a_j)$$

The conditional distributions for ancestor index variables C_k are defined by the vector parameter θ. For the first block, $Pr(c_1) = \theta_{1,c_1}$ and for subsequent blocks, $Pr(c_k|c_{k-1}) = \theta_{k,c_{k-1} \to c_k}$. The conditional distributions for ancestor sequence variables A_j are defined by the vector $\hat{a}_{k,c,j} \in B$ over blocks $k = 1 \ldots b$, ancestors $c = 1 \ldots q_k$ and sites $j = s_k \ldots e_k$. Each sub-vector $\hat{a}_{k,c}$ defines the sequence of ancestor c of block k, so that $Pr(a_j|c_k) = 1$ if $a_j = \hat{a}_{k,c_k,j}$

and 0 otherwise. Note that the conditional distribution for A_j is deterministic, as denoted by its double border in the graph. The conditional distributions for observed sequence variables H_j are given by the vector parameter $\mu_{j,a\to h}$, defined over sites $j = 1 \ldots l$ and alleles $a, h \in B$. In each case, $Pr(h_j|a_j) = \mu_{j,a_j\to h_j}$. The small dot in each vertex H_j denotes that this variable's value is observed, whereas all others must be inferred. On this point, it is worth noting the similarities between our model and a Hidden Markov Model (HMM), since in each case there is a Markov chain of distributions over unobserved variables upon which the observed data is conditional.

Many biological assumptions underlie our model's design. Most fundamentally, we assume our population is in Hardy-Weinberg equilibrium, so we define our distribution over individual haplotypes instead of genotypes [22]. The Markov chain connecting variables $C_1 \ldots C_b$ also implies that the probability of a haplotype being descended from a particular ancestor for block k depends only on their ancestor for block $k-1$. This first-order property is based upon the observation that recombination is a Markovian process, under the assumption of no chiasma interference. The values of q_k for each block k are allowed to differ, since the processes of drift and selection act somewhat independently on each block.

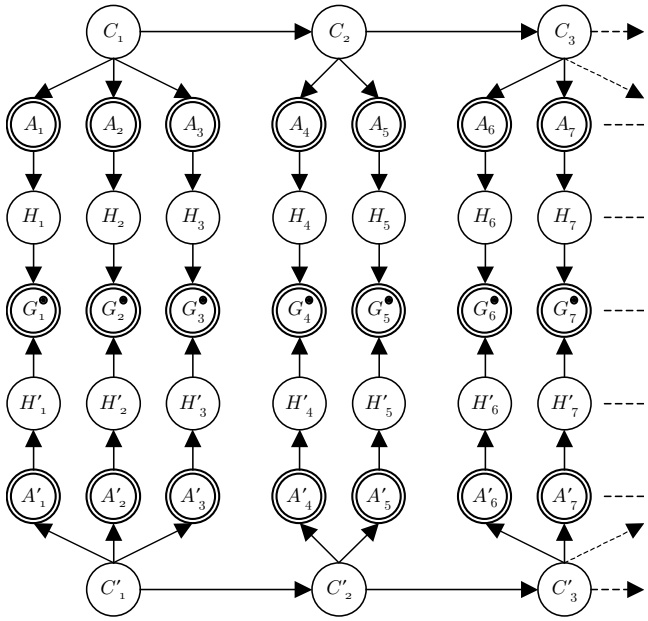
The parameter independence of each conditional distribution $Pr(A_j|C_k)$ lifts all constraints on the phylogenetic relationship between each block's ancestors, since we are only interested in tracing ancestry as far back as the formative bottleneck event. The parameter independence of each conditional distribution $Pr(H_j|A_j)$ allows for both site- and allele-specific mutation rates, justified by recent evidence for mutation hotspots [23, 24]. This is a marked departure from the traditional infinite-sites model of mutation, which assumes that each SNP has only mutated once in evolutionary history. Our model also assumes that a site's mutation rate is independent of the alleles at other sites, or the ancestor from which it is descended.

Nonetheless, a model's mutation rates are constrained in other ways. Firstly, if either a or h are not observed alleles of site j, we fix $\mu_{j,a\to h} = 0$, since such mutations are assumed either never to occur or to be deleterious. For other alleles $a \neq h$, mutation rates are constrained by parameters μ_{min} and μ_{max}, so that $\mu_{min} \leq \mu_{j,a\to h} \leq \mu_{max}$. The values of μ_{min} and μ_{max} should ideally be based on the mutability and history of the chromosomal region being studied. However, since we generally lack such knowledge, suitable guideline values are $\mu_{min} = 10^{-6}$ and $\mu_{max} = 10^{-3}$, based on mutation rates of 1.6×10^{-7} to 5.5×10^{-9} per generation, a generation length of 20 years and a most recent bottleneck event between 100,000 and 5,000 years ago [25].

The Markov chain parameters θ determine some additional values of interest. For the first block, the prior distribution $\pi_{1,c}$ for each ancestor c is clearly given by $\pi_{1,c} = \theta_{1,c}$. For subsequent blocks $k > 1$, we obtain the prior distribution from that of the previous block and the transition parameters, where $\pi_{k,c} = \sum_{c'} (\pi_{k-1,c'} \cdot \theta_{k,c'\to c})$. The conditional entropy $\xi_{(k-1)\to k}$ across each hotspot measures the degree of recombination between blocks $k-1$ and k and is given by $\xi_{(k-1)\to k} = -\sum_{c'} \pi_{k-1,c'} \sum_{c} (\theta_{k,c'\to c} \cdot \log \theta_{k,c'\to c})$.

Under a particular model M, the likelihood $Pr(h|M)$ of a haplotype $h = h_1, \ldots, h_l$ is obtained by calculating the probability of the corresponding partial assignment in the Bayesian Network. This is given by the summation of the joint probability function over all unassigned variables, i.e. $\sum_{c_1 \ldots c_b} \sum_{a_1 \ldots a_l} Pr(c_1, \ldots, c_b, a_1, \ldots, a_l, h_1, \ldots, h_l|M)$, calculated efficiently by bucket variable elimination [20]. In some cases, we lack observations for particular sites due to failed measurements in the laboratory, in which case the variables H_j corresponding to those sites are unassigned and so included in the summation.

Figure 2: Bayesian Network for genotype data



The likelihood $Pr(g|M)$ of a genotype g is calculated using the Bayesian Network shown in Figure 2. This contains two identical copies of the haplotype Bayesian Network corresponding to M, where the mirrored copy has variables renamed to C'_k, A'_j and H'_j. The new discrete variable G_j corresponds to the joint observation at site j, so we evaluate a genotype's likelihood by calculating the probability of the partial assignment $Pr(g_1, \ldots, g_l|M)$. Each G_j takes values from the set D of possible unordered pairs of SNP alleles, given by $D = \{[b_1, b_2] : b_1, b_2 \in B\}$. The conditional distribution for each G_j is deterministic, since it is fixed by the alleles present on each chromosome at site j, i.e. $Pr(g_j|h_j, h'_j) = 1$ if $g_j = [h_j, h'_j]$ and 0 otherwise.

3. MDL CRITERION

Our core problem is to learn a suitable model from observed SNP data, consisting of a set of haplotype observations H and/or genotype observations G. Assuming sample independence, the likelihood $Pr(H, G|M)$ of the data under model M is given by $\prod_{h\in H} Pr(h|M) \prod_{g\in G} Pr(g|M)$.

Seeking a model which maximizes this likelihood produces erroneous results, since any observed haplotype distribution can be reproduced exactly by a simple model with no recombination or mutation. We address this problem of model over-fitting using the minimum description length (MDL) criterion, which seeks to minimize the total number of bits required to represent data with a model, akin to finding its optimal compressed encoding [26]. If $DL(M)$

bits are required to represent a model M for data D then $DL(D,M) = DL(M) - \log_2 Pr(D|M)$. For general Bayesian Networks, the Bayesian Information Criterion (BIC) can be used to calculate $DL(M)$ but we diverge somewhat from that formulation here [27].

Formally, the description length $DL(M)$ of model M is the number of bits required to represent it with optimal efficiency. For our models, we ignore elements of this description whose lengths are fixed, for example the boolean vector describing the partition into blocks and the site mutation rates μ, since these make no difference to model comparisons. We consider only an efficient representation of the ancestor sequences $\hat{a}$ and the parameters θ of the Markov chain.

Ancestor sequences are represented using a distribution-based optimal encoding scheme [28]. First, for each SNP j, the frequency $f_j(a)$ in the model's ancestors of each allele a is calculated independently. If SNP j falls in block k, this is given by $f_j(a) = \frac{1}{q_k} |\{c : \hat{a}_{k,c,j} = a\}|$. These independent frequencies are multiplied to form a distribution over the SNPs in block k, so that $Pr(\hat{a}_{k,c}) = \prod_{j=s_k}^{e_k} f_j(\hat{a}_{k,c,j})$. Using our scheme, the representation length of the sequence of ancestor c of block k is given by $L(\hat{a}_{k,c}) = -\log_2 Pr(\hat{a}_{k,c})$, so the total length for all ancestor sequences of block k is $S_k = \sum_c L(\hat{a}_{k,c})$. Note that we ignore the cost of representing the actual allele frequencies $f_j(a)$, since this is fixed for all models to be compared.

Since each parameter θ of the Markov chain is a continuous value with potentially infinite representation size, a limit must be placed on its accuracy. We apply Rissanen's result, which states that the optimal representation size for continuous parameters of a distribution from which m samples are taken is $\frac{1}{2}\log_2 m$ bits [29]. Therefore, the cost T_1 to represent all $\theta_{1,c}$ parameters for the first block is given by $T_1 = \frac{q_1-1}{2}\log_2 n$, where $n = |H| + 2|G|$ is the number of haplotypes represented by our data. Similarly, the cost T_k to represent all $\theta_{k,c'\to c}$ parameters for subsequent blocks $k > 1$ is given by $T_k = \frac{q_k-1}{2} q_{k-1} \log_2 n$.

Thus, the total description length of a model M is given by $DL(M) = \sum_k (S_k + T_k)$ and our aim is to find M which minimizes $DL(H,G,M) = DL(M) - \log_2 Pr(H,G|M)$.

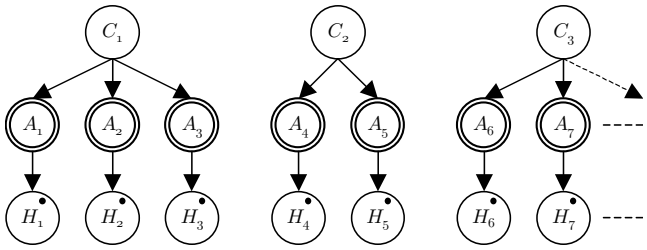
4. SEARCH ALGORITHM

Clearly, for any non-trivial input, the space of possible models is vast (to begin with, there are 2^{l-1} different partitions into blocks) so any form of exhaustive search is infeasible. Instead, our strategy takes advantage of two features of the search space which were observed during development.

Firstly, it was noted that if the optimal model has several recombination hotspots, adding these one-by-one will tend to incrementally improve the score. This means that hotspots may be examined individually and accumulated over several iterations. Secondly, even if the recombination hotspots in a model are not quite at their ideal locations or the number of ancestors for each block is slightly sub-optimal, the model will nonetheless have a relatively strong score. This means that an initial quick scan can be used to assess regions of the search space, leading to further exploration in those areas which look most promising.

Globally, we adopt a myopic search strategy, retaining and attempting to improve only the best scoring model M found to date. We begin by assigning M to an initial model containing no recombination hotspots, optimizing the number of ancestors for the single block. Following this, we repeatedly execute a set of three phases, hotspot addition, nudging and removal, replacing M as we go by any model found with a lower DL score. If two full rounds of these three phases produce no improvement, the algorithm finishes and the model is output, after its parameters are refined by additional rounds of EM.

Figure 3: Broken Bayesian Network for haplotype data



During hotspot addition, we attempt to insert a single new hotspot somewhere within each block of the current model, optimizing the number of ancestors for the new blocks generated on both sides. When nudging, we try moving each existing hotspot a small distance, also allowing small changes in the number of ancestors for the blocks on both sides. In the removing phase, we attempt to take out each existing hotspot, optimizing the number of ancestors for the newly reunited block.

Any particular assignment of hotspots and values q fixes the topology of the Bayesian Network and the cardinality of each variable within, allowing the remaining parameters $\hat{a}$, μ and θ to be inferred by the EM algorithm [21]. However, to speed up our search, we learn $\hat{a}$ and μ for each block independently, before learning parameters θ for the Markov chain from adjacent pairs of blocks. This is equivalent to performing EM on nodes A_j and H_j in the broken Bayesian Network shown in Figure 3, followed by EM on each node C_k with just the single edge from C_{k-1} to C_k reintroduced from Figure 1.

Learning in this modular fashion means that during our model search, we need only recalculate parameters of blocks which are immediately affected by each adding, nudging or removing operation. At the cost of losing some information, this shortcut introduces greater locality into our search space, reducing calculation time a great deal. For example, having added a hotspot within block k in an existing model M, we only relearn the ancestors $\hat{a}_k$ and $\hat{a}_{k+1}$, mutation rates $\mu_{s_k}, \ldots, \mu_{e_{k+1}}$ and Markov transition probabilities θ_k, θ_{k+1} and θ_{k+2}. Parameters for unaffected blocks are copied from M, shifting indices appropriately. Furthermore, to calculate the new value of $DL(H,G,M)$, the elements $S_1, \ldots, S_{k-1}, S_{k+1}, \ldots, S_b$ and $T_1, \ldots, T_{k-1}, T_{k+2}, \ldots, T_b$ can be reused, along with cached forward probabilities such as $Pr(h_{s_1}, \ldots, h_{e_{k-1}}, c_{k-1}|M)$ and backward probabilities such as $Pr(h_{s_{k+2}}, \ldots, h_{e_b}|c_{k+1}, M)$ for each input haplotype h.

Our model requires a deterministic conditional distribution for each variable A_j but the EM algorithm will rarely produce this. Therefore, when learning parameters within block k, we begin by fixing the conditional distribution for each $H_{s_k} \ldots H_{e_k}$ as if no mutations have taken place. Then

we perform EM for the variables $A_{s_k} \ldots A_{e_k}$, effectively clustering the observed sequences into q_k self-similar clades. Ancestor sequences are assigned based on each conditional distribution, setting $\hat{a}_{k,c,j} = \arg\max_a Pr(A_j = a | C_k = c)$. Only then do we perform EM for variables H_j, constraining site mutation rates to μ_{min} and μ_{max} as appropriate.

Unlike the nudging and removal phases, which examine each hotspot in the current model in turn, the addition phase requires testing every possible hotspot location within each block, significantly raising its complexity. For a new hotspot tried in block k, different numbers of ancestors q_k and q_{k+1} for the new blocks must also be considered, with only an upper limit on the likely range of suitable values. Furthermore, because the EM algorithm is guaranteed only to find parameters which lead to a local maximum for the likelihood of observed data, multiple iterations with different random seeds must be run for each assignment to q_k and q_{k+1}, in order to allow the observed sequences to be clustered best. Clearly, it would be infeasible to implement such a full search for every hotspot that could be introduced into the model.

To overcome this problem, the addition phase takes advantage of the properties of the search space, as mentioned above. The search for a suitable hotspot addition within block k takes place in two stages, called scan and isolation. In the scan stage, we generate a vector of new models V_j for each possible insertion site $j = s_k + 1 \ldots e_k - 1$, in each case copying the number of ancestors q_k and q_{k+1} in the two new blocks from q_k in the original model. Then, for each model in the vector, we try removing ancestors from each of the two new blocks in ascending order of their prior probability π, keeping any improvements in score. Having done so, the score of each model V_j is a fair guide to the value of adding a hotspot at j.

In the isolation stage, we begin by discarding all models in V whose score is lower than that of either of their neighbors. This search for local minima is guaranteed to remove at least half (rounded down) of the models remaining. Then, we try to improve each model V_j by slightly moving the newly placed hotspot and reselecting ancestors, as in the nudging phase described above. Having done so, the search for local minima is repeated, continuing the isolation process until a single model remains. In each round of the isolation stage, we double the search time expended on improving each remaining model, leading to a constant cost per round. To prevent a bias towards hotspot accumulation in early blocks, we do not attempt to add hotspots into new blocks generated by the current phase of hotspot addition.

In a similar fashion, the nudging and removing phases also focus more effort on models whose parameters are closest to the best one seen. This approach is effective because models with similar parameters tend to produce similar scores, especially when the parameters are close to optimal. Nonetheless, for best results, multiple independent runs of the search algorithm may be performed, selecting the best scoring model among those obtained.

5. HAPLOTYPE RESOLUTION

Using our approach, we perform haplotype resolution in two stages. First, we search for the best model M for observed genotype data G, as explained in section 4. We then use this model to define a function $H(g, M)$ which gives a pair of haplotypes (h, h') which is compatible with each genotype $g \in G$ and likely under M. Ideally, this function would find the assignment of $h_1, \ldots, h_l, h'_1, \ldots, h'_l$ with maximum likelihood in the model's genotype Bayesian Network, giving $\arg\max_{(h,h')} Pr(g, h, h' | M)$.

Unfortunately, computing this is infeasible, since it requires a summation over all paths through the two Markov chains to generate joint distributions over h and h' before calculating their maximal assignments, an operation with exponential complexity in terms of l. Instead, we find the joint maximum likelihood assignment of the haplotype pair $h_1, \ldots, h_l, h'_1, \ldots, h'_l$ and ancestor indices $c_1, \ldots, c_b, c'_1, \ldots, c'_b$ which is compatible with g by bucket variable elimination [20]. In doing so, we only consider the single most probable path through the Markov chain that could lead to each haplotype, analogous to applying the Viterbi algorithm on a Hidden Markov Model. This approximation is reasonable because one path is likely to give a much higher probability for a particular haplotype than the others, since mutations are rare.

6. RESULTS

Many studies of the haplotypes in particular genomic regions have been carried out over the past few years [30]. However, in most cases, the haplotypes used for the study were obtained using one of the haplotype resolution algorithms described in section 1, so they hardly form a suitable basis for a comparison of such methods. Furthermore, not all studies are based on closely-spaced SNP markers, so our block-based approach would be ineffective on the data sets obtained.

Our results are based on two sources of high-density haplotype data. Rieder *et al.* studied the gene ACE located on chromosome 17, thought to be related to cardiovascular disease, examining variation at 52 biallelic markers which extend over a genomic region of 24 kb [31]. In their paper, they obtained 22 haplotypes from 11 subjects using allele-specific PCR to ensure that ambiguous genotypes were resolved correctly [7]. Patil *et al.* undertook a full study of chromosome 21, examining variation at 24,047 SNPs over a total length of 21.7 Mb [4]. They obtained 20 haplotypes from 10 subjects by separating the two copies of each subject's chromosome using a somatic cell hybrid technique [10]. For the purposes of this comparison, we examined the five contiguous stretches of approximately 100 SNPs in chromosome 21 which extend over less than 35,000 bp.

To compare the quality of haplotype resolution, we used 10 random pairings of the true haplotypes for each region to generate genotypes, which were then passed to each algorithm for haplotype resolution. We applied our approach for three different values of μ_{min} and μ_{max} in two ways, first restricting the search to models which place all the SNPs in a single block (i.e. $b = 1$) and then allowing the block divisions to also be learned. The results are compared against those for four other methods: (i) Clark's algorithm, slightly modified to deal with unknowns [11], (ii) Our local variation of the EM algorithm which overcomes its exponential complexity [13, 14, 15], (iii) The PHASE algorithm developed by Stephens *et al.* [16], (iv) A beta version of the HAPLOTYPER algorithm developed by Niu *et al.* [17].

Table 1 compares the quality of haplotype resolution, as measured by the proportion of individuals phased incorrectly. A finer comparison, shown in Table 2, is generated by measuring the proportion of pairs of adjacent sites which are phased incorrectly relative to each other. Although the

Table 1: Mean subject error rates

Proportion of subjects[a]	C21a[b]	C21b	C21c	C21d	C21e	ACE
Clark	.8222	.7300	.5300	.7900	.8444	.5091
Local EM[c]	.5889	.3900	.1300	.5800	.5667	.3545
HAPLOTYPER[d]	.6667	–	.6000	.6000	–	.2818
PHASE	.6778	.5000	.4800	.4800	.6556	.4727
HaploBlock[e], $b = 1, \mu_{max} = 10^{-4}$	.4222	.2200	.1400	.2600	.6889	.5364
HaploBlock, $b = 1, \mu_{max} = 10^{-3}$	.4556	.2300	.1000	.3100	.6778	.5636
HaploBlock, $b = 1, \mu_{max} = 10^{-2}$	.4333	.5500	.0800	.4600	.5667	.5364
HaploBlock, $\mu_{max} = 10^{-4}$	.4556	.3400	.1200	.2800	.5667	.4818
HaploBlock, $\mu_{max} = 10^{-3}$	.4778	.3300	.1200	.3800	.6444	.6818
HaploBlock, $\mu_{max} = 10^{-2}$	.7111	.4700	.1200	.4300	.5667	.7273

[a]Sites with unknowns were excluded from the comparison.
[b]All chromosome 21 regions are from contig NT002836, over the following stretches of base pairs. a: 1262471-1292884, b: 7490174-7517009, c: 10972404-10996329, d: 13622368-13650628, e: 14999072-15030226.
[c]For Local EM and HAPLOTYPER, we took the maximum likelihood result of 20 runs.
[d]The HAPLOTYPER beta version failed on data with many unknowns – averages are for successful runs, if any.
[e]For each HaploBlock run, we set $\mu_{min} = \mu_{max}^2$.

Table 2: Mean site pairwise error rates

Proportion of pairs	C21a	C21b	C21c	C21d	C21e	ACE
Clark	.0548	.0251	.0280	.0329	.0234	.0381
Local EM	.0095	.0042	.0009	.0047	.0083	.0152
HAPLOTYPER	.0224	–	.0204	.0077	–	.0102
PHASE	.0669	.0403	.0655	.0262	.0183	.0419
HaploBlock, $b = 1, \mu_{max} = 10^{-4}$	.0052	.0011	.0007	.0014	.0161	.0100
HaploBlock, $b = 1, \mu_{max} = 10^{-3}$	.0053	.0016	.0001	.0012	.0171	.0144
HaploBlock, $b = 1, \mu_{max} = 10^{-2}$	.0036	.0074	.0006	.0027	.0116	.0185
HaploBlock, $\mu_{max} = 10^{-4}$	.0039	.0015	.0001	.0008	.0048	.0109
HaploBlock, $\mu_{max} = 10^{-3}$	.0030	.0030	.0005	.0015	.0045	.0109
HaploBlock, $\mu_{max} = 10^{-2}$	.0068	.0058	.0005	.0024	.0080	.0173

first metric is common in the literature, it forms a crude basis for comparison, since it ignores the useful information contained in a pair of haplotypes which is phased wrongly at only one site. The second metric overcomes this shortcoming and is particularly relevant if the inferred haplotypes are to be used for LD mapping, which is based on correlations between disease susceptibility and the alleles present at contiguous sites.

The first set of tests, in which the number of blocks b is fixed to 1, demonstrates the effectiveness of our ancestor and mutation model, even when the possible presence of haplotype blocks is ignored. In other words, model-based Bayesian clustering is an effective method for haplotype resolution over closely-linked SNPs. For the high-resolution data from chromosome 21, the results are compelling – our approach consistently outperforms previously published algorithms, with the exception of some cases where $\mu_{max} = 10^{-2}$. The contrast is particularly marked in the site pairwise error rates, indicating the suitability of our method for high-resolution disease mapping. Our model-based approach also obtained better results than our own Local EM algorithm with the exception of data set C21e, to be discussed further below. For the ACE data set, the results are more mixed, perhaps because the lower SNP density in that study makes it less suitable for our model.

The second set of tests, in which an unrestricted model search is performed (allowing $b \geq 1$), demonstrates the extra accuracy that is achieved by allowing recombination hotspots to be included in a model. However, for chromosome 21 data sets (a) through (d), there is no significant difference between the results of the two experiments. This surprising result is explained by the fact that even in the unrestricted model search, many of the models learned from these regions placed all the SNPs in a single block. By contrast, the unrestricted searches for data set (e) showed a clear improvement in mean site pairwise error rate from $(0.0161, 0.0171, 0.0116)$ to $(0.0048, 0.0045, 0.0080)$ for the three values of μ_{max}, reflecting the fact that they all indicated the presence of recombination hotspots. Clearly, for data that extends over longer chromosomal regions, the contrast between the two types of search will increase in prominence.

Our algorithms have been implemented in ANSI C as the HAPLOBLOCK package, available online with documentation at `http://bioinfo.cs.technion.ac.il/haploblock/`. Running times on a 2 GHz Pentium Xeon workstation were under 5 minutes for each search performed on genotype input data, while learning from haplotypes is typically 20 times faster. The search algorithm can accept a mixture of haplotypes and genotypes and imposes no limits on input size.

7. FUTURE WORK

The above results demonstrate the potential of our approach for modeling high-density SNP data. We are now expanding our study, generating a full recombination map of chromosome 21. We are also improving our search strategy to incorporate MCMC elements, to better reflect the uncertainty which is inherent in the identification of recombination hotspots. Having done this, we will perform both haplotype resolution and linkage disequilibrium mapping using the set of obtained models as a representative sample.

On a more fundamental level, our model might be improved by the introduction of prior distributions for some parameters. This is particularly relevant for mutation rates, since the alleles at adjacent sites have been observed to affect SNP mutability [25]. Similarly, we wish to test whether the first-order property of our Markov chain holds true for real data, since it might be undermined by genetic drift, local interactions between alleles or interference-like effects.

Acknowledgements

We wish to thank Perlegen for making the chromosome 21 data available and Niu *et al.* and Harvard University for the HAPLOTYPER beta. We also acknowledge the helpful input provided by A. Templeton.

8. REFERENCES

[1] Goldstein D. B. Islands of linkage disequilibrium. *Nature Genetics*, 29(2):109–11, 2001.

[2] Jeffreys A. *et al.* Intensely punctate meiotic recombination in the class II region of the major histocompatibility complex. *Nature Genetics*, 29(2):217–222, 2001.

[3] Daly M. J. *et al.* High-resolution haplotype structure in the human genome. *Nature Genetics*, 29(2):229–32, 2001.

[4] Patil N. *et al.* Blocks of limited haplotype diversity revealed by high-resolution scanning of human chromosome 21. *Science*, 294(5547):1719–23, 2001.

[5] Gabriel S. B. *et al.* The Structure of Haplotype Blocks in the Human Genome. *Science*, 296(5576):2225–9, 2002.

[6] Zhang K. *et al.* A dynamic programming algorithm for haplotype block partitioning. *PNAS USA*, 99(11):7335–9, 2002.

[7] Michalatos-Beloin S. *et al.* Molecular haplotyping of genetic markers 10 kb apart by allele-specific long-range PCR. *Nucleic Acids Research*, 24(23):4841–3, 1996.

[8] Woolley A. T. *et al.* Direct haplotyping of kilobase-size DNA using carbon nanotube probes. *Nature Biotechnology*, 18(7):760–3, 2000.

[9] Lizardi P. M. *et al.* Mutation detection and single-molecule counting using isothermal rolling-circle amplification. *Nature Genetics*, 19(3):225–32, 1999.

[10] Douglas J. A. *et al.* Experimentally-derived haplotypes substantially increase the efficiency of linkage disequilibrium studies. *Nature Genetics*, 28(4):361–4, 2001.

[11] Clark A. G. Inference of haplotypes from PCR-amplified samples of diploid populations. *Molecular Biology and Evolution*, 7(2):111–22, 1990.

[12] Gusfield D. Inference of haplotypes from samples of diploid populations: complexity and algorithms. *Journal of Computational Biology*, 8(3):305–23, 2001.

[13] Excoffier L. & Slatkin M. Maximum-likelihood estimation of molecular haplotype frequencies in a diploid population. *Molecular Biology and Evolution*, 12(5):921–7, 1995.

[14] Long J. C. *et al.* An E-M algorithm and testing strategy for multiple-locus haplotypes. *American Journal of Human Genetics*, 56(3):799–810, 1995.

[15] Templeton A. R. A cladistic analysis of phenotypic associations with haplotypes inferred from restriction endonuclease mapping. II. The analysis of natural populations. *Genetics*, 120:1145–1154, 1988.

[16] Stephens M. *et al.* A new statistical method for haplotype reconstruction from population data. *American Journal of Human Genetics*, 68(4):978–89, 2001.

[17] Niu T. *et al.* Bayesian haplotype inference for multiple linked single-nucleotide polymorphisms. *American Journal of Human Genetics*, 70(1):157–69, 2002.

[18] Pearl J. *Probabilistic Reasoning in Intelligent Systems*. Morgan Kaufmann, San Mateo, CA, 2nd edition, 1988.

[19] Jensen F. V. *An Introduction to Bayesian Networks*. Springer Verlag, New York, NY, 1996.

[20] Dechter R. Bucket elimination: A unifying framework for probabilistic inference. In *Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence (UAI-96)*, pages 211–219, August 1–4 1996.

[21] Lauritzen S. L. The EM algorithm for graphical association models with missing data. *Computational Statistics and Data Analysis*, 19:191–201, 1995.

[22] G. H. Hardy. Mendelian proportions in a mixed population. *Science*, 18:49–50, 1908.

[23] Templeton A. R. *et al.* Recombinational and mutational hotspots within the human lipoprotein lipase gene. *American Journal of Human Genetics*, 66(1):69–83, 2000.

[24] Fullerton S. *et al.* Apolipoprotein E variation at the sequence haplotype level: implications for the origin and maintenance of a major human polymorphism. *American Journal of Human Genetics*, 67(4):881–900, 2000.

[25] Nachman M.W. & Crowell S.L. Estimate of the mutation rate per nucleotide in humans. *Genetics*, 156(1):297–304, 2000.

[26] Rissanen J. Modeling by shortest data description. *Automatica*, 14:465–471, 1978.

[27] Schwarz, G. Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–4, 1978.

[28] Shannon C. E. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423, 623–656, 1948.

[29] Rissanen J. A universal prior for integers and estimation by minimum description length. *Annals of Statistics*, 11:416–431, 1983.

[30] Ardlie K.G. *et al.* Patterns of linkage disequilibrium in the human genome. *Nature Reviews Genetics*, 3(7):299–309, 2002.

[31] Rieder M. J. *et al.* Sequence variation in the human angiotensin converting enzyme. *Nature Genetics*, 22(1):59–62, 1999.